

## QUERY REWRITING WITH ENTITY DETECTION

### BACKGROUND OF THE INVENTION

#### Field of the Invention

[0001] Systems and methods consistent with the principles of the invention relate generally to information retrieval and, more particularly, to rewriting of search queries based on detection of the names of certain entities in the queries.

#### Description of Related Art

[0002] The World Wide Web ("web") contains a vast amount of information. Search engines assist users in locating desired portions of this information by cataloging web documents. Typically, in response to a user's request, a search engine returns links to documents relevant to the request.

[0003] Search engines may base their determination of the user's interest on search terms (called a search query) provided by the user. The goal of a search engine is to identify links to relevant results based on the search query. Typically, the search engine accomplishes this by matching the terms in the search query to a corpus of pre-stored web documents. Web documents that contain the user's search terms are considered "hits" and are returned to the user.

[0004] Some search engines permit a user to restrict a search to a set of related documents, such as documents associated with the same web site, by including special characters or terms in the search query. Oftentimes, however, users forget to include these special characters/terms or do not know about them.

SUMMARY OF THE INVENTION

[0005] According to one aspect consistent with the principles of the invention, a method may include receiving a search query, determining whether the received search query includes an entity name, and determining whether the entity name is associated with a common word or phrase. The method may also include selectively rewriting the received search query based on whether the entity name is determined to be associated with a common word or phrase, performing a search based on the received search query or the rewritten search query to obtain search results, and presenting the search results.

[0006] According to another aspect, a system may include means for receiving a search query, means for determining whether the received search query includes an entity name, and means for determining whether the entity name is associated with a common word or phrase. The system may also include means for rewriting the received search query when it is determined that the entity name is associated with a common word or phrase, means for performing a search based on the rewritten search query to obtain search results, and means for providing the search results.

[0007] According to yet another aspect, a system includes a memory and a processor connected to the memory to receive a search query, determine whether the received search query includes an entity name, and selectively rewrite the received search query to obtain a rewritten search query when it is determined that the received search query includes an entity name.

[0008] According to a further aspect, a method may include determining a set of entity names, determining whether each of the entity names is associated with a common word or phrase, and generating a table of the entity names that are associated with common words or phrases.

[0009] According to another aspect, a method may include receiving a search query, determining whether the received search query includes an entity name, and determining whether the entity name is associated with a common word or phrase. When the entity name is associated with a common word or phrase, the method may include generating a link to a rewritten query, performing a search based on the received search query to obtain first search results, and providing the first search results and the link to the rewritten query. When the entity name is not associated with a common word or phrase, the method may include rewriting the received search query to include a restrict identifier associated with the entity name, generating a link to the received search query, performing a search based on the rewritten search query to obtain second search results, and providing the second search results and the link to the received search query.

#### BRIEF DESCRIPTION OF THE DRAWINGS

[0010] The accompanying drawings, which are incorporated in and constitute a part of this specification, illustrate an embodiment of the invention and, together with the description, explain the invention. In the drawings,

[0011] Fig. 1 is a diagram of an exemplary network in which systems and methods consistent with the principles of the invention may be implemented;

[0012] Fig. 2 is an exemplary diagram of a client and/or server of Fig. 1 according to an implementation consistent with the principles of the invention;

[0013] Fig. 3 is an exemplary functional block diagram of a portion of a server of Fig. 1 according to an implementation consistent with the principles of the invention;

[0014] Fig. 4 is an exemplary diagram of a list of candidate strings according to an implementation consistent with the principles of the invention;

[0015] Fig. 5 is a flowchart of exemplary processing for generating a list of candidate strings according to an implementation consistent with the principles of the invention;

[0016] Fig. 6 is a flowchart of exemplary processing for selectively rewriting a query according to an implementation consistent with the principles of the invention;

[0017] Figs. 7 and 8 are diagrams of an automatic query rewrite example in a news context according to an implementation consistent with the principles of the invention; and

[0018] Figs. 9-11 are diagrams of a query rewrite suggestion example in the news context according to an implementation consistent with the principles of the invention.

### DETAILED DESCRIPTION

[0019] The following detailed description of the invention refers to the accompanying drawings. The same reference numbers in different drawings may identify the same or similar elements. Also, the following detailed description does not limit the invention.

### OVERVIEW

[0020] Systems and methods consistent with the principles of the invention may rewrite search queries or generate suggestion links to rewritten search queries upon detection of the names of certain entities. An "entity," as used herein, may refer to anything that can be tagged as being associated with certain documents. Examples of entities may include news sources, stores, such as online stores, product categories, brands or manufacturers, specific product models, condition (e.g., new, used, refurbished, etc.), authors, artists, people, places, and organizations.

[0021] Some entity names are unambiguous and uniquely identify particular entities. A large number of names, however, are somewhat ambiguous or generic, making it more difficult to identify the entities to which they are intended to correspond when included in users' search

queries. Systems and methods consistent with the principles of the invention provide mechanisms for determining the entities to which entity names correspond and selectively rewriting users' search queries based on the entity names. Accordingly, a user's search query may be restricted to a search of document(s) associated with the entity that the user intended in the search.

#### EXEMPLARY NETWORK CONFIGURATION

[0022] Fig. 1 is an exemplary diagram of a network 100 in which systems and methods consistent with the principles of the invention may be implemented. Network 100 may include multiple clients 110 connected to multiple servers 120-140 via a network 150. Network 150 may include a local area network (LAN), a wide area network (WAN), a telephone network, such as the Public Switched Telephone Network (PSTN), an intranet, the Internet, a memory device, or a combination of networks. Two clients 110 and three servers 120-140 have been illustrated as connected to network 150 for simplicity. In practice, there may be more or fewer clients and servers. Also, in some instances, a client may perform the functions of a server and a server may perform the functions of a client.

[0023] Clients 110 may include client components. A component may be defined as a device, such as a wireless telephone, a personal computer, a personal digital assistant (PDA), a lap top, or another type of computation or communication device, a thread or process running on one of these devices, and/or an object executable by one of these device. Servers 120-140 may include server components that gather, process, search, and/or maintain documents in a manner consistent with the principles of the invention. Clients 110 and servers 120-140 may connect to network 150 via wired, wireless, and/or optical connections.

[0024] In an implementation consistent with the principles of the invention, server 120 may include a search engine 125 usable by clients 110. Server 120 may crawl a corpus of documents (e.g., web pages), index the documents, and store information associated with the documents in a repository of crawled documents. Servers 130 and 140 may store or maintain documents that may be crawled by server 120. While servers 120-140 are shown as separate entities, it may be possible for one or more of servers 120-140 to perform one or more of the functions of another one or more of servers 120-140. For example, it may be possible that two or more of servers 120-140 are implemented as a single server. It may also be possible for a single one of servers 120-140 to be implemented as two or more separate (and possibly distributed) devices.

[0025] A "document," as the term is used herein, is to be broadly interpreted to include any machine-readable and machine-storable work product. A document may include an e-mail, a web site, a file, a combination of files, one or more files with embedded links to other files, a news group posting, a blog, a web advertisement, etc. In the context of the Internet, a common document is a web page. Web pages often include textual information and may include embedded information (such as meta information, images, hyperlinks, etc.) and/or embedded instructions (such as Javascript, etc.). A "link," as the term is used herein, is to be broadly interpreted to include any reference to or from a document.

#### EXEMPLARY CLIENT/SERVER ARCHITECTURE

[0026] Fig. 2 is an exemplary diagram of a client or server component (hereinafter called "client/server component"), which may correspond to one or more of clients 110 and servers 120-140, according to an implementation consistent with the principles of the invention. The client/server component may include a bus 210, a processor 220, a main memory 230, a read only memory (ROM) 240, a storage device 250, an input device 260, an output device 270, and a

communication interface 280. Bus 210 may include a path that permits communication among the elements of the client/server component.

[0027] Processor 220 may include a conventional processor or microprocessor, or another type of processing logic that interprets and executes instructions. Main memory 230 may include a random access memory (RAM) or another type of dynamic storage device that stores information and instructions for execution by processor 220. ROM 240 may include a conventional ROM device or another type of static storage device that stores static information and instructions for use by processor 220. Storage device 250 may include a magnetic and/or optical recording medium and its corresponding drive.

[0028] Input device 260 may include a conventional mechanism that permits an operator to input information to the client/server component, such as a keyboard, a mouse, a pen, voice recognition and/or biometric mechanisms, etc. Output device 270 may include a conventional mechanism that outputs information to the operator, including a display, a printer, a speaker, etc. Communication interface 280 may include any transceiver-like mechanism that enables the client/server component to communicate with other devices and/or systems. For example, communication interface 280 may include mechanisms for communicating with another device or system via a network, such as network 150.

[0029] As will be described in detail below, the client/server component, consistent with the principles of the invention, may perform certain searching-related operations. The client/server component may perform these operations in response to processor 220 executing software instructions contained in a computer-readable medium, such as memory 230. A computer-readable medium may be defined as a physical or logical memory device and/or carrier wave.

[0030] The software instructions may be read into memory 230 from another computer-readable medium, such as data storage device 250, or from another device via communication interface 280. The software instructions contained in memory 230 may cause processor 220 to perform processes that will be described later. Alternatively, hardwired circuitry may be used in place of or in combination with software instructions to implement processes consistent with the principles of the invention. Thus, implementations consistent with the principles of the invention are not limited to any specific combination of hardware circuitry and software.

#### EXEMPLARY SERVER

[0031] Fig. 3 is an exemplary functional block diagram of a portion of server 120 according to an implementation consistent with the principles of the invention. According to one implementation, one or more of the functions described below may be performed by search engine 125. According to another implementation, one or more of these functions may be performed by a component external to server 120, such as a computer associated with server 120 or one of servers 130 and 140.

[0032] Server 120 may include an entity identification unit 310 and a query processing unit 320 connected to a repository. The repository may include information associated with documents that were previously crawled and stored, for example, by server 120.

[0033] Entity identification unit 310 may generate a list of entity names. Entity identification unit 310 may obtain an initial set of entity names for entities in a particular context (e.g., names of news sources in the news source context or store names in the store context). There are many ways that entity identification unit 310 can obtain the initial set of entity names in a particular context. For example, entity identification unit 310 may obtain entity names from online directories, lists, group postings, by analyzing a corpus of documents, etc.



[0034] For each of these names, entity identification unit 310 may also identify an entity identifier, such as a homepage domain name or a category identifier, associated with the name. For example, if the name was Washington Post, then the associated entity identifier might be washingtonpost.com. Entity identification unit 310 may identify the associated entity identifier from, for example, an analysis of the document information in the repository.

[0035] Entity identification unit 310 may then process the entity names to produce a list of variations of the names. Entity identification unit 310 may apply several transformations to the name and/or its entity identifier, such as: using the entity name as is; using the entity identifier as is; removing modifiers, such as "a," "the," "inc," "inc.," "co," and "co." from the entity name; replacing spaces with hyphens or underscores, or vice versa, within the entity name; removing apostrophes from the entity name; interchanging "and" and "&" in the entity name and/or the entity identifier; removing "and" and "&" from the entity name and/or the entity identifier; removing the initial "www." and/or the trailing ".com" from the entity identifier; and/or treating periods in the entity identifier with no spaces on either side of them as spaces or deleting the periods. Other or different transformations may also be used.

[0036] Entity identification unit 310 may form these name variations into a list of candidate strings. Fig. 4 is an exemplary diagram of a list of candidate strings 400 according to an implementation consistent with the principles of the invention. Candidate string list 400 might include a number of entries (candidate strings) associated with the various versions of entity names and their associated entity identifiers. An entry in list 400 might include an entity name field 410 and an entity ID field 420. Entity name field 410 may include a variation of an entity name or its associated entity identifier. Entity ID field 420 may include information that uniquely identifies the entity corresponding to the entity name in entity name field 410, such as a

domain, a URL, or a category identifier. An example of an entry for the news source Washington Post might include "washington post" in entity name field 410 and "www.washingtonpost.com" in entity ID field 420.

[0037] Returning to Fig. 3, query processing unit 320 may process the list of candidate strings to determine whether a search query should be automatically rewritten or whether rewriting of a query should be suggested. For example, query processing unit 320 may determine whether a query includes an entity name or any variation thereof. Query processing unit 320 may check the terms of the query against list of candidate strings 400 (Fig. 4). In one implementation, query processing unit 320 may check whether a word, or phrase (hereinafter "term" will be used to encompass both a "word" and a "phrase"), at the left or right most position of the query matches one of the candidate strings. In another implementation, query processing unit 320 may check whether any term in the query matches one of the candidate strings.

[0038] If a term matches one of the candidate strings, query processing unit 320 may optionally determine whether a word in the query that neighbors the term indicates that no further processing of the query should occur. For example, query processing unit 320 may determine whether a word that neighbors the term (e.g., is adjacent to or near the term) forms a common phrase with the term, such that the combination of this word with the term forms a phrase that should not be decomposed.

[0039] To illustrate this, assume that the query includes the words "time travel" and the term "time" has been identified as an entity name. The user who provided the query may have meant two things. First, the user may want to find information on the phrase "time travel." Alternatively, the user may want to find information on "travel" from the news source "Time."

In this case, query processing unit 320 may recognize the phrase "time travel" as a common phrase and determine that the phrase should not be decomposed.

[0040] Query processing unit 320 may identify common phrases from an exhaustive list of phrases. The list of phrases may be obtained from a number of sources. One such source may include the repository of documents. For example, documents in the repository may be analyzed to identify phrases that appear more than a threshold number of times in different documents.

[0041] When query processing unit 320 determines that no further processing of the query should occur, then query processor 320 may perform a search using the original query and present the search results to the user. In this case, query processing unit 320 may optionally include a link to a rewritten query with the search results. The rewritten query may restrict the search to the entity identifier (e.g., domain) associated with the entity name (or variation) in the query.

[0042] When query processing unit 320 determines that further processing of the query should occur, then query processing unit 320 may determine whether the term is associated with a common word or phrase. There are several ways that query processing unit 320 may determine whether the term is associated with a common word or phrase. For example, query processing unit 320 may compare the term to a dictionary of English words and phrases. Alternatively, query processing unit 320 may use an inverse document frequency (IDF) weighting technique or a conventional linguistic modeling technique. One such technique may involve analyzing a corpus of documents and creating a hash table based on the terms in the documents. For example, each term in a document may be identified and hashed. The count value in the corresponding entry in the hash table may then be incremented. Once the corpus has been analyzed, the count values may reflect which terms occurred more often and which terms

occurred less often. Query processing unit 320 may identify terms that have occurred more than a threshold amount as common terms.

[0043] If query processing unit 320 determines that the query term is not associated with a common word or phrase, then query processing unit 320 may rewrite the query. The rewritten query may be based on the identification of an entity name and restrict the query to a search associated with the entity name. For example, if a user query includes "washingtonpost," then the query may be rewritten to "source:washingtonpost" to indicate that the search is to be restricted to the entity identifier (domain) associated with the news source Washington Post. The "source:" may correspond to a restrict identifier in the news context that indicates that the search should be restricted to the news source that follows it. Similar restrict identifiers may be used in other contexts.

[0044] Query processing unit 320 may then perform a search based on the rewritten query and present results to the user. Query processing unit 320 may also offer a query link associated with the original query to the user. The query link, if selected by the user, may cause query processing unit 320 to perform a search based on the original query (i.e., without restricting the search to a particular entity).

[0045] If query processing unit 320 determines that the query term is associated with a common word or phrase, then query processing unit 320 may use the original query to perform a search (i.e., without restricting the search to a particular entity). Query processing unit 320 may also generate a query link associated with a rewritten query. Query processing unit 320 may rewrite the query, as described above, and provide a link to this rewritten query to the user. The query link, if selected by the user, may cause query processing unit 320 to perform a search based on the rewritten query.

## EXEMPLARY PROCESSING

[0046] Fig. 5 is a flowchart of exemplary processing for generating a list of candidate strings according to an implementation consistent with the principles of the invention. Processing may begin with obtaining a list of entity names for a particular context (act 510). For each of the entity names, a corresponding entity identifier may also be identified (act 520). Several techniques exist for identifying entity names and/or entity identifiers for the list. For example, entity names and/or entity identifiers may be identified from online directories, lists, group postings, by analyzing a corpus of documents, etc.

[0047] A list of candidate strings may then be produced by transforming the entity names and/or entity identifiers (act 530). For example, the list of candidate strings for a particular entity name and its associated entity identifier may include the entity name as is, the entity identifier as is, the entity name without modifiers (e.g., "a," "the," "inc," "inc.," "co," and "co."), the entity name with spaces replaced with hyphens or underscores, and vice versa, the entity name without apostrophes, the entity name and/or entity identifier with "and" replaced with "&," and vice versa, the entity name and/or entity identifier without "and" and "&," the entity identifier without an initial "www." and/or a trailing ".com," and the entity identifier with a period with no spaces on either side of it replaced with spaces or deleted. Other or different transformations may also be used. One such list of candidate strings is illustrated in Fig. 4.

[0048] Fig. 6 is a flowchart of exemplary processing for selectively rewriting a search query according to an implementation consistent with the principles of the invention. Processing may begin with receiving a search query from a user (act 610). The search query may contain one or more terms, which may or may not include the name of an entity.

[0049] The search query may be evaluated to identify possible entity names based on the list of candidate strings (act 620). For example, a term of the search query may be compared to the entity names, which include the variations of the entity names, in the list of candidate strings. In one implementation, the terms at the left-most position and/or right-most position within the search query may be evaluated to determine whether they correspond to one of the entity names in the list of candidate strings. In another implementation, each term of the query may be evaluated.

[0050] If a term in the search query matches one of the entity names, it may then optionally be determined whether the search query should be further processed (act 630). For example, it may be determined whether a word in the search query that neighbors the entity name forms a common phrase with the entity name, such that the combination of this word with the entity name forms a phrase that should not be decomposed. Common phrases may be identified from an exhaustive list of phrases, as described above.

[0051] When it is determined that no further processing of the query should occur, such as when a word in the search query forms a common phrase with the entity name, a search using the original query may be performed and the search results presented to the user. Optionally, a link to a rewritten query may be presented with the search results. The rewritten query may restrict the search to the entity identifier (e.g., domain) associated with the entity name in the query.

[0052] When it is determined that further processing of the query should occur, then it may be determined whether the entity name is associated with a common word or phrase (act 640). For example, the entity name may be compared to a dictionary of English words and phrases to determine whether it is associated with a common word or phrase. Alternatively, an IDF weighting technique or a conventional linguistic may be used, as described above.

[0053] In one implementation, portions of act 640 may be performed beforehand to generate a table of entity names that are common words or phrases. In this case, the determination of whether the entity name is associated with a common word or phrase may be performed by a simple table lookup operation.

[0054] If it is determined that the entity name is not associated with a common word or phrase, then the query may be rewritten to restrict the query to a search associated with the entity name (act 650). For example, the query may be rewritten to include a restrict identifier associated with a particular context. The restrict identifier may thereby restrict a search associated with the query to a search associated with the entity name. A search may then be performed based on the rewritten query.

[0055] A query link may also be generated that links to the original query (i.e., without restricting the search to a particular entity name) (act 660). The query link may be beneficial in those instances where the user did not intend a search based on the rewritten query.

[0056] If it is determined that the entity name is associated with a common word or phrase, then a query link to a rewritten query may be generated (act 670). For example, the query may be rewritten, as described above. Selection of the query link by the user may cause a search to be performed based on the rewritten query. A search may then be performed using the original query (i.e., without restricting the search to a particular entity name) (act 680).

[0057] The search, which may be performed based on the rewritten query, if applicable, or the original query, if applicable, may identify documents that are relevant to the rewritten/original query. For example, a repository of documents may be searched to identify documents that include one or more terms of the query. The resulting documents may form

search results that may be presented to the user (act 690). In one implementation, the search results might take the form of links to the documents.

#### AUTOMATIC QUERY REWRITE EXAMPLE - NEWS CONTEXT

[0058] Figs. 7 and 8 are diagrams of an automatic query rewrite example in the news context according to an implementation consistent with the principles of the invention. As shown in Fig. 7, a user may enter a search query via a graphical user interface associated with a search engine, such as search engine 125 (Fig. 1). In this example, the user enters the search query "george bush msnbc." Assume that the term "msnbc" identifies the news source msnbc.com and, thus, is included in the list of candidate strings (e.g., see Fig. 4).

[0059] Search engine 125 may identify "msnbc" as an entity name. Assume that search engine 125 determines that the phrase "bush msnbc" and/or the phrase "george bush msnbc" are not common phrases. Search engine 125 may then evaluate the entity name "msnbc" to determine whether it is associated with a common word or phrase. In this case, search engine 125 determines that "msnbc" is not associated with a common word or phrase. Search engine 125 may then rewrite the query to "george bush source:msnbc," as shown in Fig. 8.

[0060] Search engine 125 performs a search of a repository for documents (e.g., news documents) associated with the source msnbc.com that are relevant to the rewritten query. There are many ways to determine document relevancy. For example, documents that contain one or more of the search terms of the rewritten query may be identified as relevant. Documents that include a greater number of the search terms may be identified as more relevant than documents that include a fewer number of the search terms.

[0061] Search engine 125 may then present the relevant documents to the user as search results. As shown in Fig. 8, each search result may include a link 810 to a corresponding



document, a news source identifier along with an indicator of when the document was created 820, and a brief description 830 of the corresponding document. Search engine 125 may also provide a query link 850 to the original query entered by the user. In this case, query link 850 may correspond to a query associated with a search for the search term "george," the search term "bush," and/or the search term "msnbc."

#### SUGGEST QUERY REWRITE EXAMPLE - NEWS CONTEXT

[0062] Figs. 9-11 are diagrams of a query rewrite suggestion example in the news context according to an implementation consistent with the principles of the invention. As shown in Fig. 9, a user may enter a search query via a graphical user interface associated with a search engine, such as search engine 125 (Fig. 1). In this example, the user enters the search query "time korea." Assume that the term "time" identifies the news source time.com and, thus, is included in the list of candidate strings (e.g., see Fig. 4).

[0063] Search engine 125 may identify "time" as an entity name. Assume that search engine 125 determines that the phrase "time korea" is not a common phrase. Search engine 125 may then evaluate the entity name "time" to determine whether it is associated with a common word or phrase. In this case, search engine 125 determines that "time" is associated with a common word or phrase. Search engine 125 may then rewrite the query to "korea source:time" and generate a link 1010 ("Search News Source Time for Korea") to the rewritten query, as shown in Fig. 10.

[0064] Search engine 125 performs a search of a repository for documents (e.g., news documents) that are relevant to the original search query. As described above, there are many ways to determine document relevancy. For example, documents that contain one or more of the search terms of the rewritten query may be identified as relevant. Documents that include a

greater number of the search terms may be identified as more relevant than documents that include a fewer number of the search terms. In this case, search engine 125 searches for documents that include the search terms "time" and/or "korea."

[0065] Search engine 125 may then present the relevant documents to the user as search results. As shown in Fig. 10, each search result may include a link 1020 to a corresponding document, a news source identifier along with an indicator of when the document was created 1030, and a brief description 1040 of the corresponding document. Because the search was not limited to the news source Time, the search results are associated with a number of different news sources (e.g., the New York Times, British Broadcasting Corporation (BBC), and Atlanta Journal Constitution).

[0066] If the user selects link 1010 associated with the rewritten query, search engine 125 performs a search of the repository for documents (e.g., news documents) associated with the news source time.com that are relevant to the rewritten query. Search engine 125 may then present the relevant documents to the user as search results. As shown in Fig. 11, each search result may include a link 1110 to a corresponding document, a news source identifier along with a date indicator 1120 corresponding to the date on which the document was created, and a brief description 1130 of the corresponding document. Optionally, search engine 125 may also provide a link 1150 to the original query entered by the user. In this case, link 1150 may correspond to a query associated with a search for the search term "time" and/or the search term "korea."

## CONCLUSION

[0067] Systems and methods consistent with the principles of the invention may selectively rewrite search queries upon detection of the names of certain entities.

[0068] The foregoing description of preferred embodiments of the present invention provides illustration and description, but is not intended to be exhaustive or to limit the invention to the precise form disclosed. Modifications and variations are possible in light of the above teachings or may be acquired from practice of the invention.

[0069] For example, it has been described that query processing unit 320 may perform a search based on the original or rewritten search query. In other implementations, query processing unit 320 may not perform the search, but may provide the original or rewritten search query to a search engine, such as search engine 125 (Fig. 1) to perform the search and provide the search results.

[0070] Also, while series of acts have been described with regard to Figs. 5 and 6, the order of the acts may be modified in other implementations consistent with the principles of the invention. Further, non-dependent acts may be performed in parallel.

[0071] In one implementation, server 120 may perform most, if not all, of the acts described with regard to the processing of Figs. 5 and/or 6. In another implementation consistent with the principles of the invention, one or more, or all, of the acts may be performed by another component, such as another server 130 and/or 140 or client 110.

[0072] It will also be apparent to one of ordinary skill in the art that aspects of the invention, as described above, may be implemented in many different forms of software, firmware, and hardware in the implementations illustrated in the figures. The actual software code or specialized control hardware used to implement aspects consistent with the principles of the invention is not limiting of the present invention. Thus, the operation and behavior of the aspects were described without reference to the specific software code--it being understood that one of

ordinary skill in the art would be able to design software and control hardware to implement the aspects based on the description herein.

[0073] No element, act, or instruction used in the present application should be construed as critical or essential to the invention unless explicitly described as such. Also, as used herein, the article "a" is intended to include one or more items. Where only one item is intended, the term "one" or similar language is used. Further, the phrase "based on" is intended to mean "based, at least in part, on" unless explicitly stated otherwise.